What is claimed is:

1. A method for managing file security attributes by a file server in a computer file storage system, the method comprising:

receiving a first request from a Windows client relating to a file stored in the computer file storage system;

determining that the file is a UNIX-secured file;

retrieving a set of UNIX file security attributes associated with the file, the set of UNIX file security attributes including at least a UNIX owner identifier and a UNIX group identifier; and

generating a set of Windows file security attributes from the set of UNIX file security attributes, the set of Windows file security attributes including a plurality of security identifiers (SID) including at least an owner SID derived from the UNIX owner identifier and a group SID derived from the UNIX group identifier, wherein at least one of the owner SID and the group SID includes at least one UNIX-specific indicator and the corresponding UNIX identifier.

2. A method according to claim 1, wherein the at least one UNIX-specific indicator includes a UNIX-specific authority identifier having a value other than the well-known authority identifiers zero through five and an owner/group indicator having a first value to indicate that the UNIX identifier is the UNIX owner identifier and a second value to indicate that the UNIX identifier is the UNIX group identifier.

3. A method according to claim 1, wherein the at least one UNIX-specific indicator includes a UNIX-specific authority identifier having a first value other than the well-known authority identifiers zero through five to indicate that the UNIX identifier is the UNIX owner identifier and a second value other than the well-known authority identifiers zero through five to indicate that the UNIX identifier is the UNIX group identifier.

4. A method according to claim 1, wherein generating a set of Windows file security attributes from the set of UNIX file security attributes comprises:

attempting to map each UNIX identifier to a corresponding Windows identifier; and

generating, for each UNIX identifier that cannot be mapped to a corresponding Windows identifier, the SID including the at least one UNIX-specific indicator and the corresponding UNIX identifier.

5. A method according to claim 4, wherein attempting to map each UNIX identifier to a corresponding Windows identifier comprises:

maintaining a table mapping UNIX names to Windows names;

determining a UNIX name corresponding to the UNIX identifier; and

searching the table for a Windows name corresponding to the UNIX name.

6. A method according to claim 5, wherein determining a UNIX name corresponding to the UNIX identifier comprises:

maintaining a cache mapping UNIX identifiers to UNIX names; and

searching the cache for a UNIX name corresponding to the UNIX identifier.

7. A method according to claim 5, wherein determining a UNIX name corresponding to the UNIX identifier comprises:

sending the UNIX identifier over a communication link to a NIS server; and

receiving the UNIX name over the communication link from the NIS server.

8. A method according to claim 1, further comprising:

transmitting the set of Windows file security attributes to the Windows client in a response to the first request.

9. A method according to claim 8, further comprising:

receiving a second request from the Windows client including at least one of said SIDs including at least one UNIX-specific indicator and the corresponding UNIX identifier;

translating the at least one of said SIDs into a text string; and

transmitting the text string to the Windows client in a response to the second request.

10. A method according to claim 9, wherein the text string includes a representation of the UNIX identifier from the SID.

11. A method according to claim 1, wherein the set of UNIX file security attributes includes a set of UNIX file permissions, and wherein generating the set of Windows file security attributes from the set of UNIX file security attributes further comprises:

generating a set of Windows file permissions from the set of UNIX file permissions.

12. A method according to claim 11, wherein the request comprises at least one requested change to the security attributes of the file, and wherein the method further comprises:

applying the requested security attribute changes to the set of Windows file security attributes to create a modified set of Windows file security attributes; and

writing the modified set of Windows file security attributes to the file, said writing effectively changing the file from UNIX-secured to Windows-secured.

13. A method according to claim 12, further comprising:

receiving a second request from a UNIX client relating to the file, the second request associated with a session, the session having a session owner and a session group;

retrieving the modified set of Windows file security attributes for the file; and

providing the UNIX client with owner access to the file, if the owner SID in the modified set of Windows file security attributes includes a UNIX owner identifier and the session owner matches the UNIX owner identifier in the owner SID.

14. A method according to claim 12, further comprising:

receiving a second request from a UNIX client relating to the file, the second request associated with a session, the session having a session owner and a session group;

retrieving the modified set of Windows file security attributes for the file; and

providing the UNIX client with group access to the file, if the group SID in the modified set of Windows file security attributes includes a UNIX group identifier and the session group matches the UNIX group identifier in the group SID.

15. A method according to claim 11, wherein generating the set of Windows file permissions from the set of UNIX file permissions comprises:

translating the set of UNIX file permissions into a set of Windows file permissions, the set of Windows file permissions defining owner permissions, group permissions, and everyone permissions;

removing any rights from the owner that the owner would be granted implicitly but are not granted to either the group or to everyone;

adding any rights that need to be explicitly denied to the owner and to the group;

producing a set of access control elements ordered hierarchically; and

removing any redundant permissions from the access control elements.

16. An apparatus for managing file security attributes in a computer file storage system, the apparatus comprising:

a network interface for communicating with clients over a communication network;

a storage interface for communicating with a file storage device; and

file security logic operating between the network interface and the storage interface for managing file security attributes, the file security logic including logic for generating a set of Windows file security attributes from the set of UNIX file security attributes, the set of Windows file security attributes including at least an owner SID derived from the UNIX owner identifier and a group SID derived from the UNIX group identifier, wherein at least one of the owner SID and the group SID includes at least one UNIX-specific indicator and the corresponding UNIX identifier.

17. An apparatus according to claim 16, wherein the at least one UNIX-specific indicator includes a UNIX-specific authority identifier having a value other than the well-known authority identifiers zero through five and an owner/group indicator having a first value to indicate that the UNIX identifier is the UNIX owner identifier and a second value to indicate that the UNIX identifier is the UNIX group identifier.

18. An apparatus according to claim 16, wherein the at least one UNIX-specific indicator includes a UNIX-specific authority identifier having a first value other than the well-known authority identifiers zero through five to indicate that the UNIX identifier is the UNIX owner identifier and a second value

other than the well-known authority identifiers zero through five to indicate that the UNIX identifier is the UNIX group identifier.

19. An apparatus according to claim 16, wherein the file security logic comprises:

logic for mapping each UNIX identifier to a corresponding Windows identifier; and

logic for generating, for each UNIX identifier that cannot be mapped to a corresponding Windows identifier, the SID including the at least one UNIX-specific indicator and the corresponding UNIX identifier.

20. An apparatus according to claim 19, further comprising a table mapping UNIX names to Windows names, the file security logic determining a UNIX name corresponding to the UNIX identifier and searching the table for a Windows name corresponding to the UNIX name for mapping each UNIX identifier to a corresponding Windows identifier.

21. An apparatus according to claim 20, further comprising a cache mapping UNIX identifiers to UNIX names, the file security logic searching the cache for a UNIX name corresponding to the UNIX identifier for determining a UNIX name corresponding to the UNIX identifier.

22. An apparatus according to claim 20, wherein the file security logic sends the UNIX identifier over a communication link to a NIS server for determining a UNIX name corresponding to the UNIX identifier.

23. An apparatus according to claim 16, wherein the file security logic further comprises:

logic for translating the at least one of said SIDs into a text string.

24. An apparatus according to claim 23, wherein the text string includes a representation of the UNIX identifier from the SID.

25. A method according to claim 16, wherein the set of UNIX file security attributes includes a set of UNIX file permissions, and wherein the file security logic further comprises:

logic for generating a set of Windows file permissions from the set of UNIX file permissions.

26. An apparatus according to claim 25, wherein the file security logic includes logic for receiving a request from a Windows client to modify file security attributes, applying the requested medications to the set of Windows file permissions, and writing the modified set of Windows file permissions to the storage device.

27. An apparatus according to claim 25, wherein the file security logic includes logic for controlling access to the file using the set of Windows file permissions.

28. An apparatus according to claim 25, wherein the file security logic includes logic for translating the set of UNIX file permissions into a set of Windows file permissions, the set of Windows file permissions defining owner permissions, group permissions, and everyone permissions; removing any rights from the owner that the owner would be granted implicitly but are not granted to either the group or to everyone; adding any rights that need to be explicitly denied to the owner and to the group; producing a set of access control elements ordered hierarchically; and removing any redundant permissions from the access control elements.

29. An apparatus for managing file security attributes in a computer file storage system, the apparatus comprising:

means for translating a Unix owner identifier into a Windows-compatible owner SID;

means for translating a Unix group identifier into a Windows-compatible owner SID; and

means for translating Unix file access permissions into a Windows-compatible access control list.

30. A method for generating a set of Windows file permissions from a set of UNIX file permissions, the method comprising:

translating the set of UNIX file permissions into a set of Windows file permissions, the set of Windows file permissions defining owner permissions, group permissions, and everyone permissions;

removing any rights from the owner that the owner would be granted implicitly but are not granted to either the group or to everyone;

adding any rights that need to be explicitly denied to the owner and to the group;

producing a set of access control elements ordered hierarchically; and

removing any redundant permissions from the access control elements.

31. A method comprising:

receiving a security identifier (SID) including at least one UNIX-specific indicator and a corresponding UNIX identifier; and

translating the SID into a text string.

32. A method according to claim 31, wherein the text string includes a representation of the UNIX identifier from the SID.

33. A method according to claim 31, wherein translating the SID into a text string comprises:

transmitting a request to a translator over a communication network, the request including at least the UNIX identifier.